PROJECT REPORT

UNIVERSITY ADMIT PREDICTOR

Team ID : PNT2022TMID20967

Submitted By

MONISHA M	**412419106042**
ASHLIN SL	**412419106010**
AARTHI R	**412419106001**
LEKHASHREE J	**412419106039**
HARINI KA	**412419106026**

ABSTRACT

This project aims to build a model that can help students to pick the right universities based on their profiles. For the accurate predictions we plan on training a machine learning model in order to provide results. The dataset contains information on the student profile and the university details with a field detailing if the admission was positive or not. Various algorithms have been used i.e. Ensemble Machine Learning and the predictions have been compared using key performance indicators(KPIs). Themodel performing the best is then used to evaluate the dependent variable i.e. The chances of admit to a university. The chances of admit variable is a variable ranging from 0 to 1 which equates to the predicted probability of successful acceptance to a university. We also aim to create a portal which filters and then provides a list of universities that fall into the profile's acceptance range.

1. INTRODUCTION

1.1. Objective :

Students are often worried about their chances of admission to University. The aim of this project is to help students in shortlisting universities with their profiles. The predicted output gives them a fair idea about their admission chances in a particular university. This analysis should also help students who are currently preparing or will be preparing to get a better idea.

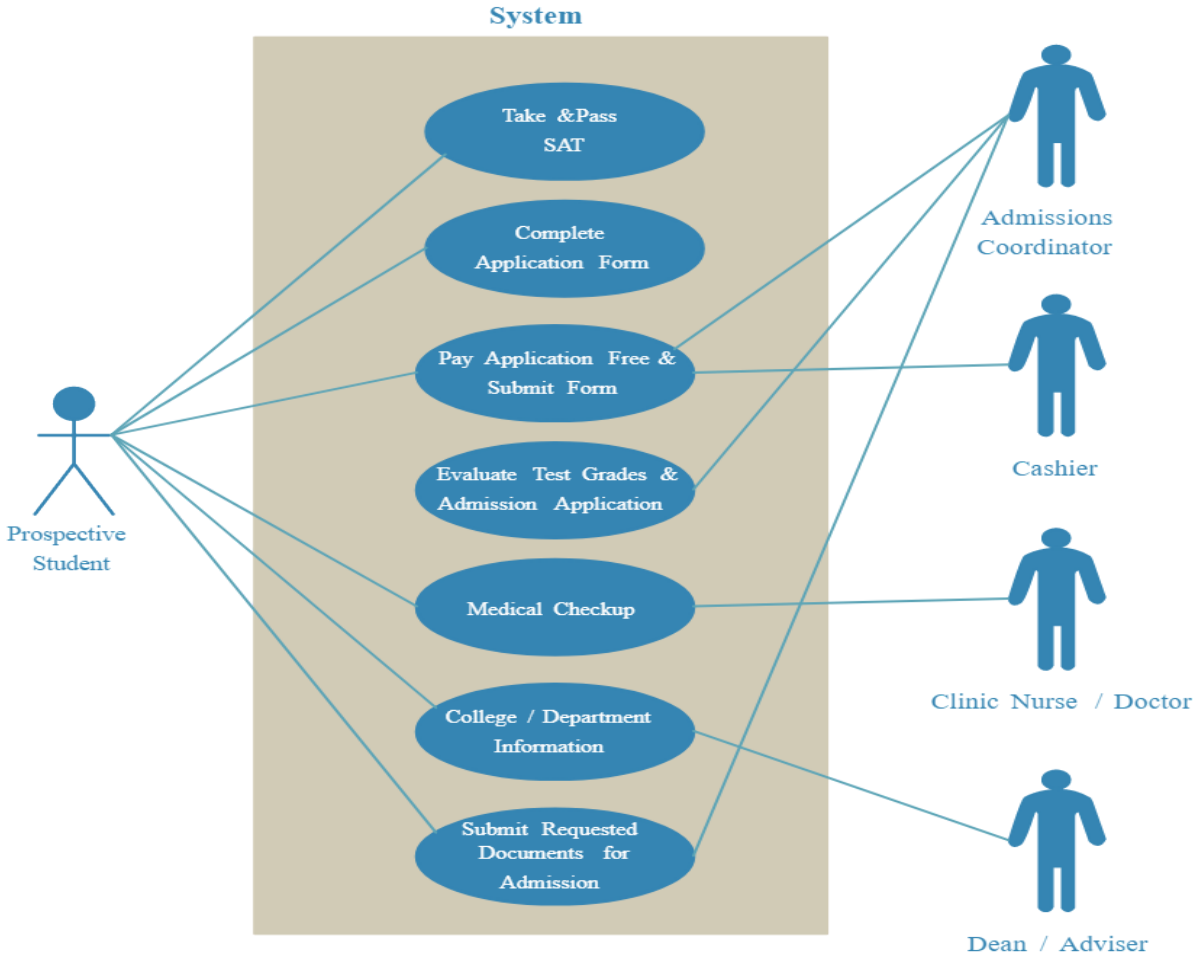
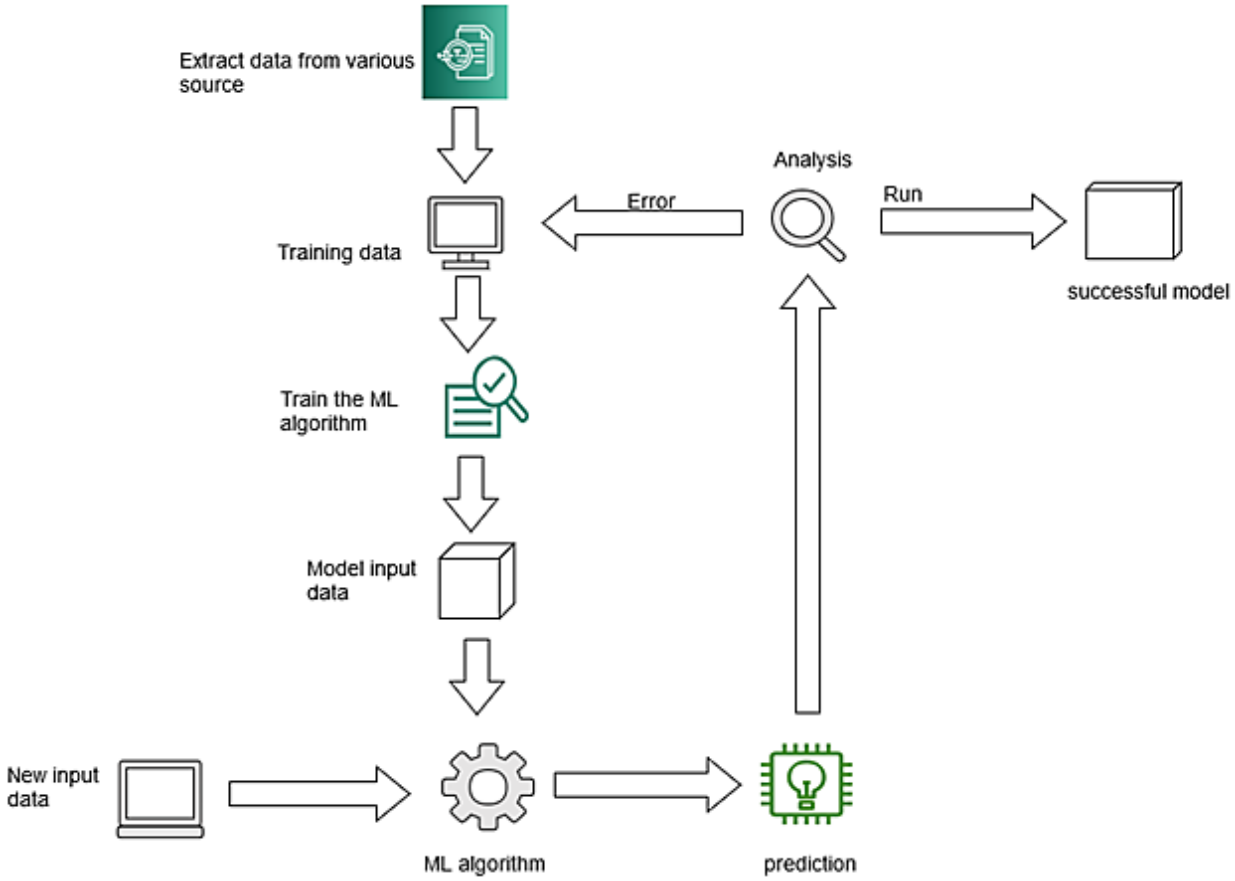


fig (i) Admission process

1.2 Problem Statement:

Educational organizations have always played an important and vital role in society for development and growth of any individual. There are different college prediction apps and websites being maintained contemporarily, but using themis tedious tosome extent,due to the lack of articulate information regarding colleges, and the time consumed in searching the best deserving college. The problem statement, hence being tackled, is to design a college prediction/prediction system and to provide a probabilistic insight into college administration for overall rating, cut-offs of the colleges, admission intake and preferences of students. Also, it helps students avoid spending time and money on counsellor and stressful research related to finding a suitable college. It has always been a troublesome process for students in finding the perfect university and course for their further studies. At times they do know which stream they want to get into, but it is not easy for them to find colleges based on their academic marks and other performances. We aim to develop and provide a place which would give a probabilistic output as to how likely it is to get into a university given upon their details**.**

1.3 Project flow in machine learning:



1.4 Learnings :

- Data Preprocessing is vital to the accuracy of the models
- Choosing appropriate machine learning techniques and algorithms to model the system
- Graphical representation of the data provides useful insights and can lead to better models
- Defining scope with respect to the dataset

2. LITERATURE SURVEY

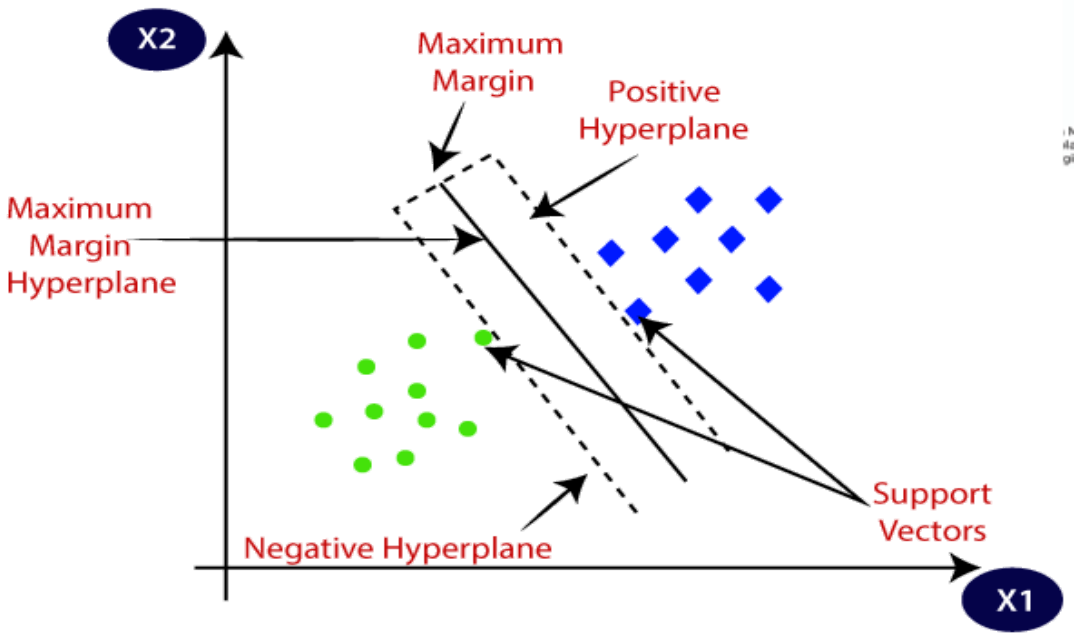
S. NO	REFERENCE PAPER	AUTHOR	ALGORITHM USED	INFERENCE
1	A University Admission Prediction System using Stacked Ensemble Learning	Sashank Sridhar, Siddhartha, Mootha	MULTI LAYER PERCEPTRON AND THE STACKED ENSEMBLE MODEL.	The proposed ensemble neural network is evaluated by comparing it to other supervised algorithms such as DecisionTrees, Random Forest, K-Nearest Neighbor, Naive Bayes Classifier, Logistic Regression, Support Vector Machine, (SVM), Linear Discriminant Analysis and Quadratic Discriminant Analysis.
2	Prediction of the admission lines of college entrance examination based on machine	Zhenru,Wang Yijie Shi Zhenru Wang, Yijie Shi	ADABOOST ALGORITHM	The proposed ensemble neural network is calculated by decision tree, random forest, K nearest neighbor and Naive Bayes Classifier Algorithms
3	Research on Prediction of College Students Performance based on Support Vector Machine	Peng Wang, Yinshan Jia	Support VECTOR MACHINE was used to establish a college course performance prediction model, and cross-validation methods were used to obtain the best parameters and a reliable and stable model.	The prediction accuracy rate reached 73.6%. The prediction result.
4	Multi-Split Optimized Bagging Ensemble Model Selection for Multi-class Educational Data Mining.	M. Injadat, A. Moubayed	KNN ALGORITHM	The prediction accuracy rate reached 78%.

3. MODEL ANALYSIS

3.1 Available Algorithms :

A. SVM Algorithm:

We can predict the admission chances by using machine learning.We can use SVM algorithm to predict the admission condition.Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.



B. Naive Bayes algorithm:

We can train our model and make it to be executed to all kinds of scenarios to achieve that we can go for Naive Bayes algorithm.Naïve Bayes algorithm is a supervised learning algorithm, which is based on Bayes theorem and used for solving classification problems.It is mainly used in text classification that includes a high-dimensional training dataset. Naïve Bayes Classifier is one of the simple and most effective classification algorithms which helps in building the fast machine learning models that can make quick predictions.It is a probabilistic classifier, which means it predicts on the basis of the probability of an object.Some popular examples of Naïve Bayes Algorithm are spam filtration, Sentimental analysis, and classifying articles.

Bayes' Theorem

- Bayes' theorem is also known as Bayes' Rule or Bayes' law, which is used to determine the probability of a hypothesis with prior knowledge. It depends on the conditional probability.
- The formula for Bayes' theorem is given as:

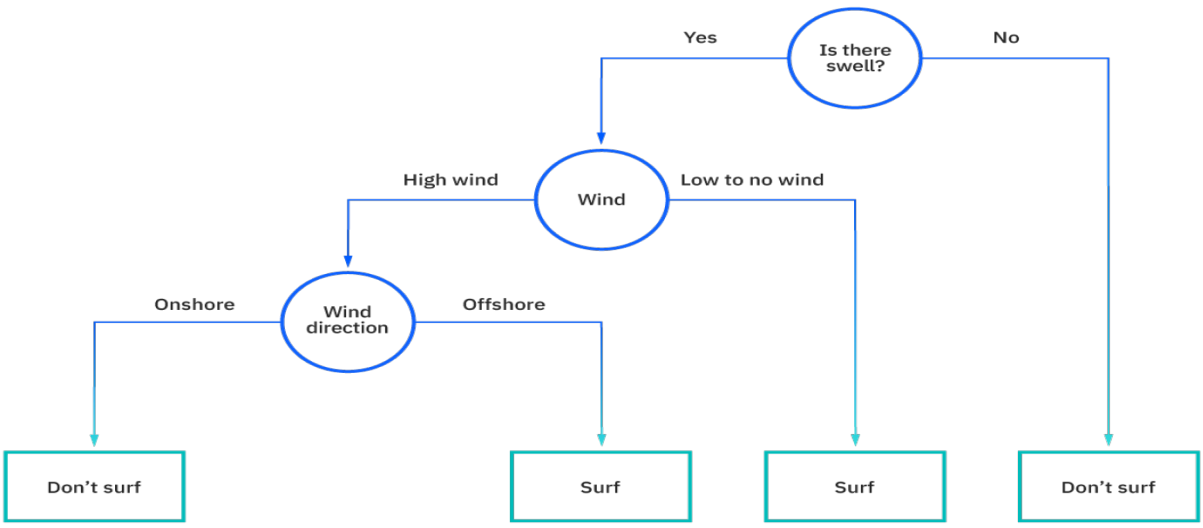
$$P(A|B)= \frac{P(B|A)P(A)}{P(B)}$$ Where,

P(A|B) is Posterior probability: Probability of hypothesis A on the observed event B.

P(B|A) is Likelihood probability: Probability of the evidence given that the probability of a hypothesis is true.

C. Decision tree algorithm:

We can use the Decision tree algorithm to train our model.A decision tree is a non-parametric supervised learning algorithm, which is utilized for both classification and regression tasks. It has a hierarchical tree structure, which consists of a root node, branches, internal nodes and leaf nodes. A decision tree starts with a root node, which does not have any incoming branches. The outgoing branches from the root node then feed into the internal nodes, also known as decision nodes. Based on the available features, both node types conduct evaluations to form homogenous subsets, which are denoted by leaf nodes, or terminal nodes. The leaf nodes represent all the possible outcomes within the dataset. As an example, let's imagine that you were trying to assess whether or not you should go surf, you may use the following decision rules to make a choice:



D. K-Means algorithm:

To obtain the predicted results we can use K-means algorithm.K-Means Clustering is an unsupervised learning algorithm that is used to solve the clustering problems in machine learning or

data science.

Working of K-Means algorithm

Step-1: Select the number K to decide the number of clusters.

Step-2: Select random K points or centroids. (It can be different from the input dataset).

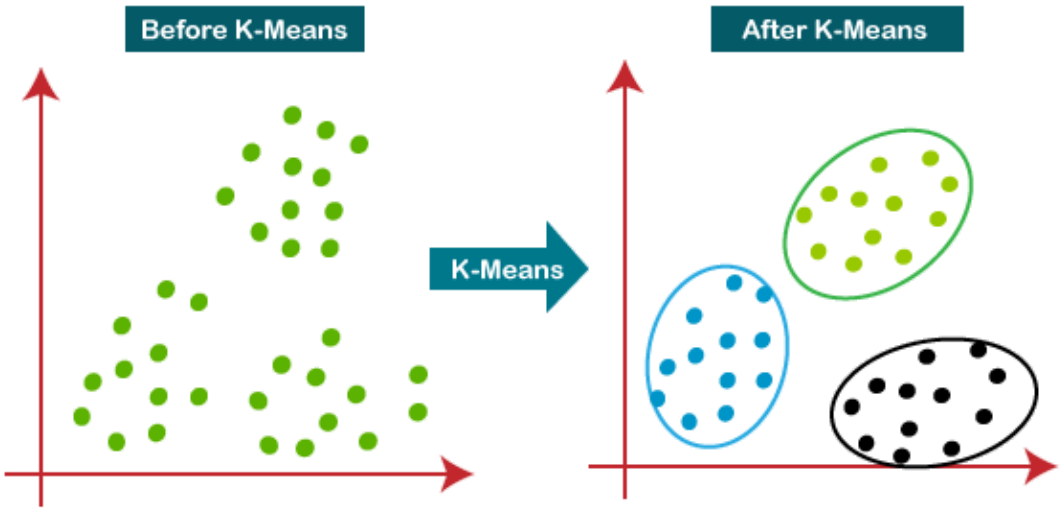
Step-3: Assign each data point to their closest centroid, which will form the predefined K clusters.

Step-4: Calculate the variance and place a new centroid of each cluster.

Step-5: Repeat the third steps, which means re-assign each datapoint to the new closest centroid of each cluster.

Step-6: If any reassignment occurs, then go to step-4 else go to FINISH.

Step-7: The model is ready.



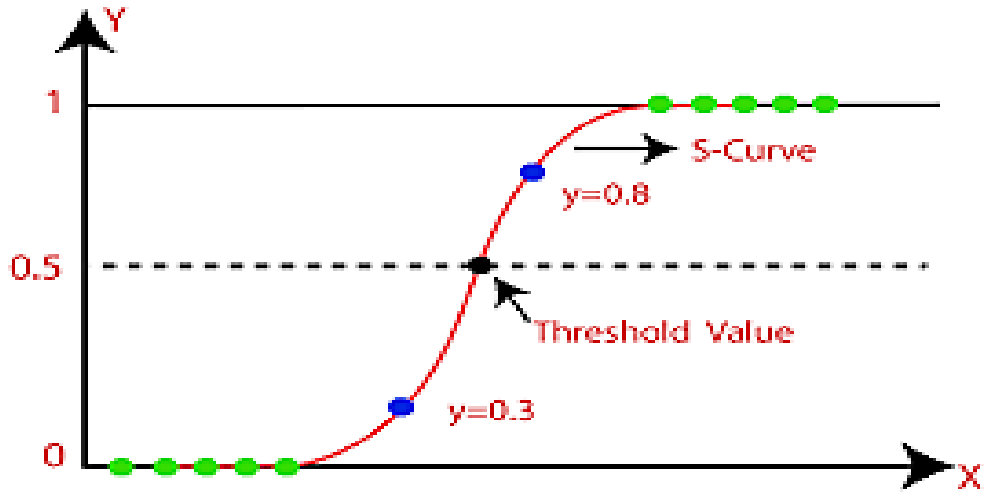
E. Logistic regression:

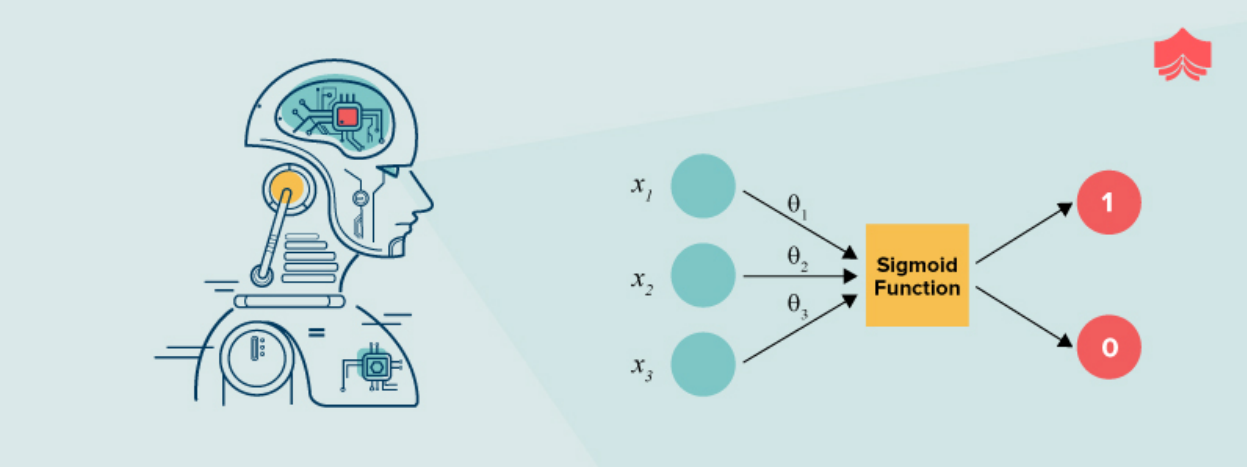
To find the best and optimal solution to the selected problem statement that is the university admission predictor we can give the solution using machine learning. In machine learning there are lots of algorithms to predict the data and to give the output based on it's prediction.This prediction must be done in a correct manner and the output must be generated as per the correct prediction.The result should have a high accuracy.To achieve that we can use logistic regression algorithm which have high accuracy almost equivalent to 85%.

Logistic regression is one of the most popular Machine Learning algorithms, which comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable using a given set of independent variables.Logistic regression predicts the output of a categorical dependent variable. Therefore the outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, True or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.

Logistic Regression is much similar to Linear Regression except that how they are used. Linear Regression is used for solving Regression problems, whereas Logistic regression is used for solving the classification problems.Logistic Regression is a significant machine learning algorithm because it has the ability to provide probabilities and classify new data using continuous and discrete datasets.

Logistic Regression can be used to classify the observations using different types of data and can easily determine the most effective variables used for the classification. The below image is showing the logistic function:





Assumptions for Logistic Regression:

- The dependent variable must be categorical in nature.
- The independent variable should not have multicollinearity

Logistic Regression Equation:

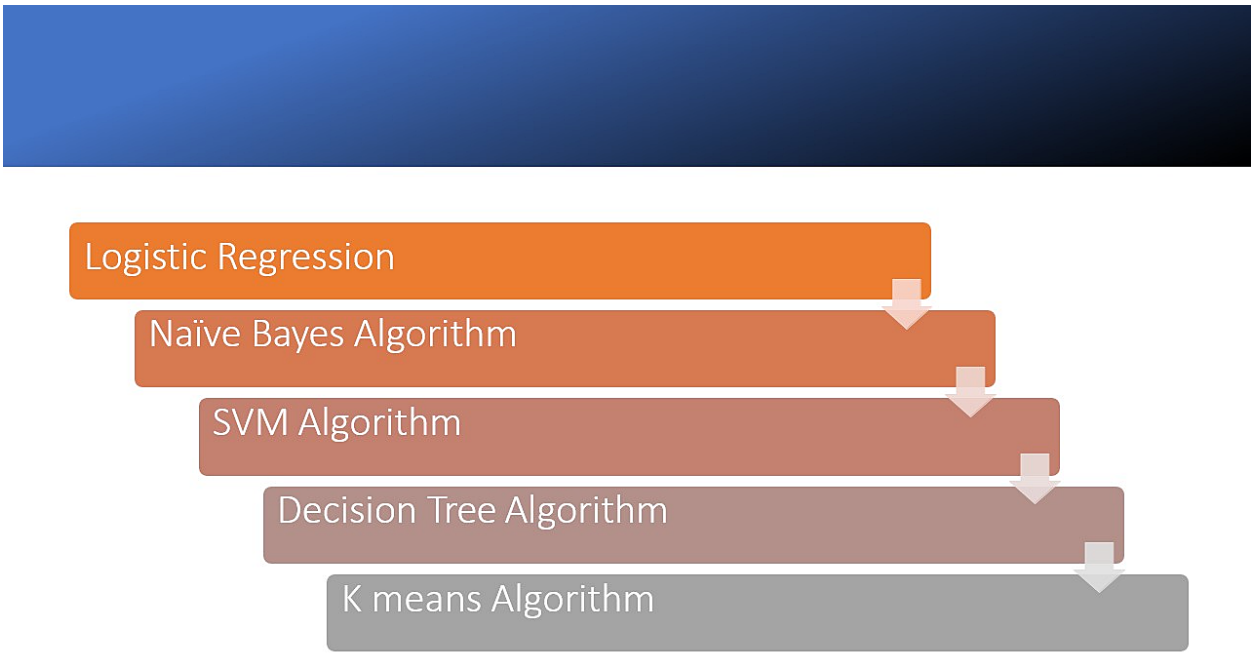
.

$$y = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \cdots + b_nx_n$$

$\frac{y}{1-y}$; 0 for y= 0, and infinity for y=1

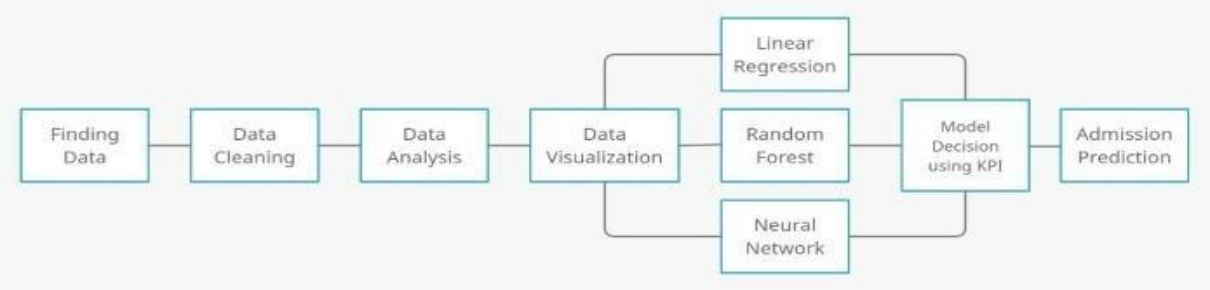
$$log\left[\frac{y}{1-y}\right] = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \cdots + b_nx_n$$

3.2 Idea Prioritization:



From this prioritization from top to bottom we decided to go with a Logistic regression algorithm to find the accurate output from our model.Because it has the highest accuracy compared with all other algorithms.It will give most accurate results among all other algorithms.

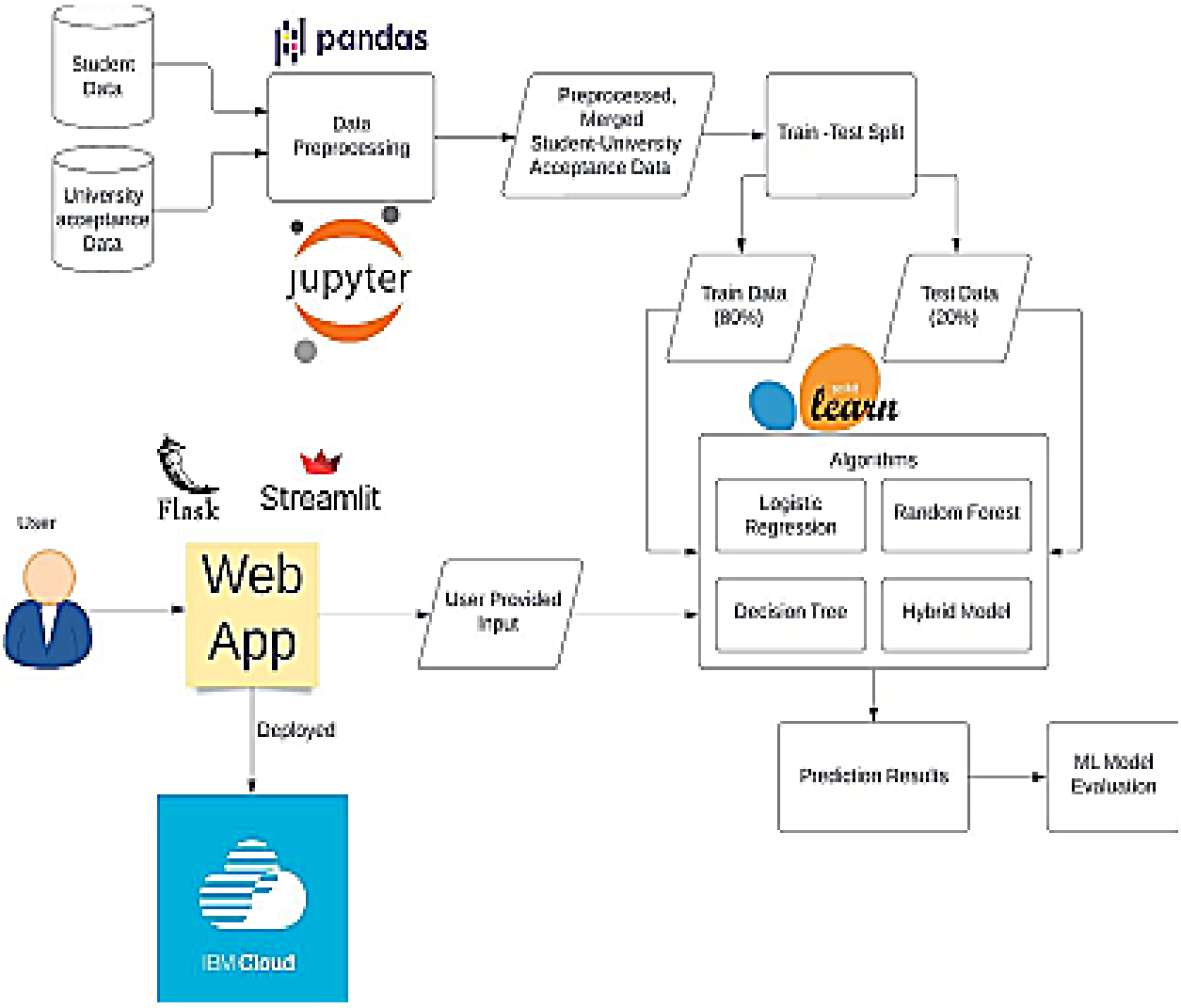
4. METHODOLOGY AND IMPLEMENTATION



The primitive step to building a model for our use case is choosing the right dataset. For our predictions, we chose a dataset which contains all the important attributes that would affect the chances of admit. This is followed by data cleaning where we handle missing values present in various fields. Once the data is ready to be analyzed, we use various tools and libraries to visualize the data and perform analysis. This includes visualizing bar graphs and the correlation matrix.

Once the data is ready to be processed, we split it into training and testing data. For this, we will be using 3 machine learning algorithms; linear regression, random forest and neural network. Once these models are built over the dataset, we compare them using key performance indicators. These indicators help us choose the right model for predicting whether an applicant has chances of admission.

4.1. Technical Architecture Diagram:



4.2. Empathy Map :

5. PROPOSED SOLUTION

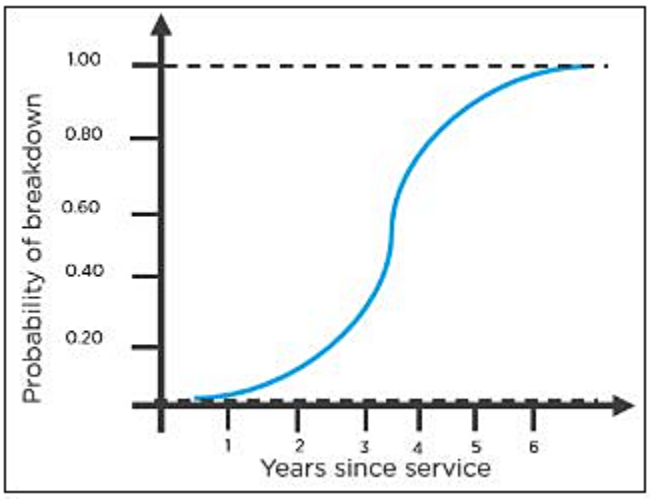
The most important part of the machine learning model is to give the high accuracy based on the given and collected data. To get the highest accuracy we have to select the appropriate algorithm which will predict the event with high accuracy. In our project we are going to use Logistic Regression Algorithm for prediction.Our model will be trained and tested by the Logistic Regression Algorithm which will give the accuracy near to 94%.

To find the best and optimal solution to the selected problem statement that is the university admission predictor we can give the solution using machine learning. In machine learning there are lots of algorithms to predict the data and to give the output based on it's prediction.This prediction must be done in a correct manner and the output must be generated as per the correct prediction.The result should have a high accuracy.To achieve that we can use logistic regression algorithm which have high accuracy almost equivalent to 85%.

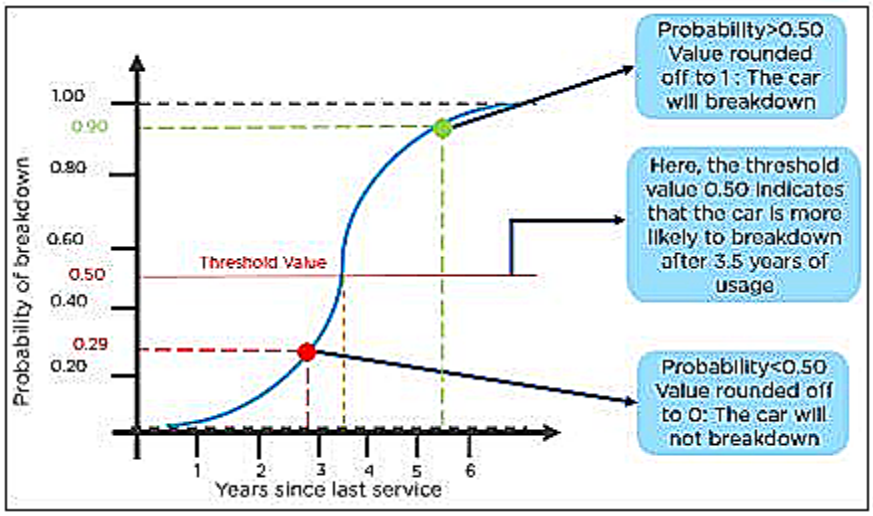
Logistic regression is a statistical method that is used for building machine learning models where the dependent variable is dichotomous: i.e. binary. Logistic regression is used to describe data and the relationship between one dependent variable and one or more independent variables. The independent variables can be nominal, ordinal, or of interval type.

The name "logistic regression" is derived from the concept of the logistic function that it uses. The logistic function is also known as the sigmoid function. The value of this logistic function lies between zero and one.

The following is an example of a logistic function we can use to find the probability of a vehicle breaking down, depending on how many years it has been since it was serviced last.



Here is how you can interpret the results from the graph to decide whether the vehicle will break down or not.



5.1 Advantages of the proposed solution :

1. Logistic regression is easier to implement, interpret, and very efficient to train.
2. It makes no assumptions about distributions of classes in feature space.
3. It can easily extend to multiple classes(multinomial regression) and a natural probabilistic view of class predictions.
4. It not only provides a measure of how appropriate a predictor(coefficient size)is, but also its direction of association (positive or negative).
5. It is very fast at classifying unknown records.
6. Good accuracy for many simple data sets and it performs well when the dataset is linearly separable.
7. It can interpret model coefficients as indicators of feature importance.
8. Logistic regression is less inclined to over-fitting but it can overfit in high dimensional datasets.One may consider Regularization (L1 and L2) techniques to avoid over-fittingin these scenarios.

6. MODEL WORKING

6.1 Dataset:

The data set comprises of different factors attributed towards picking the right university. It contains data of 100 different students.

Data set is classified into 9 different parameters which are considered important during the application for Masters.

Those parameters are: gre scores, toefl scores, university rating, statement of purpose, letter of recommendation, undergraduate gpa, research paper, chance of admit.

# Serial No.	# GRE Score	# TOEFL Score	# University Rating
1 – 400	290 – 340	92 – 120	1 – 5
1	337	118	4
2	324	107	4
3	316	104	3
4	322	110	3
5	314	103	2
6	330	115	5
7	321	109	3
8	308	101	2
9	302	102	1
10	323	108	3

Fig -1: Data set containing Gre score, Toelf score and university ranking

# SOP	# LOR	# CGPA	# Research
1 5	1 5	6.8 9.92	0 1
4.5	4.5	9.65	1
4	4.5	8.87	1
3	3.5	8	1
3.5	2.5	8.67	1
2	3	8.21	0
4.5	3	9.34	1
3	4	8.2	1
3	4	7.9	0
2	1.5	8	0
3.5	3	8.6	0

Fig-2: Dataset containing SOP,LOR,CGPA,Research

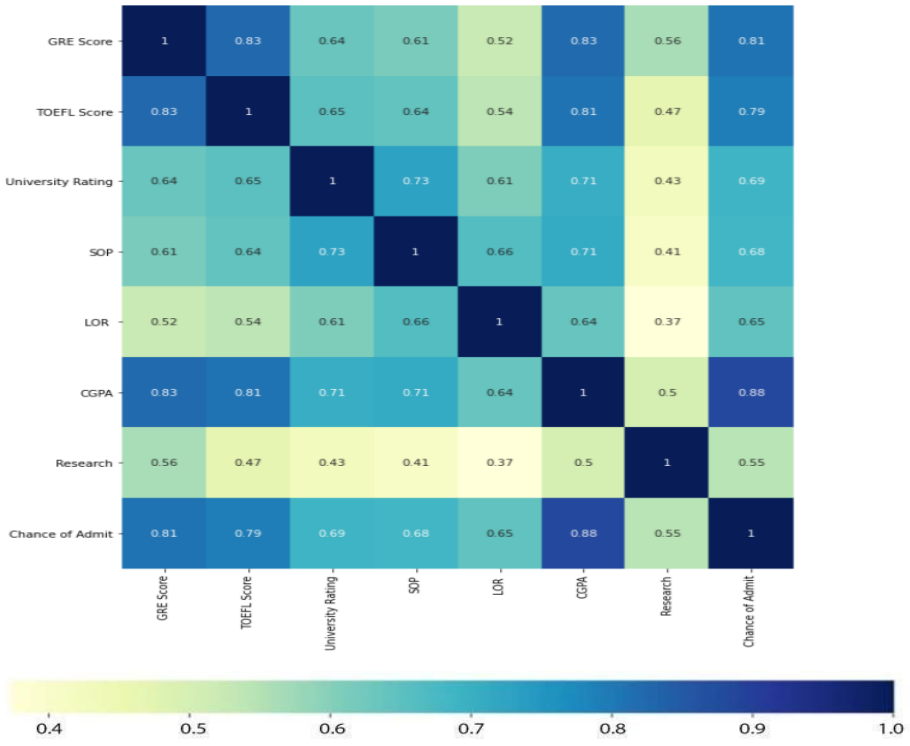


Fig-3 Correlation Matrix

6.2 Algorithm:

Step 1: Start the program.

Step 2: Importing the required libraries which have the predefined functions which will be used later during the execution.

Step 3: Getting the dataset and importing the dataset.

Step 4: Analysing the data.

Step 5: Handling the missing values.

Step 6: Splitting dependent and independent columns.

Step 7: Performing data visualization.

Step 8: Splitting 80% of data for training and 20% of data for testing.

Step 9: Calculating the accuracy, Recall score and Confusion matrix.

6.3 Project flow with code:

****IMPORTING THE LIBRARIES****

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
import os
from sklearn.preprocessing import MinMaxScaler
from sklearn.ensemble import RandomForestRegressor
import xgboost as xgb
from sklearn.metrics import mean_squared_error,
r2_score,mean_absolute_error
from sklearn import metrics
!pip install sklearn
```

****Reading the dataset****

```
data = pd.read_csv("Admission_Predict.csv")
data.head()
```

****Analyzing the data****

```
data.drop(['Serial No.'],axis=1,inplace=True)
data.head()
```

data.describe()

data.info()

Handling missing values

data.isnull().sum()

Splitting dependent and Independent columns

```
X=data.iloc[:,:6].values
X.shape

y=data['Chance of Admit ']
y.shape
```

Data visualization

```
plt.scatter(data['GRE Score'],data['CGPA'])
plt.title('CGPA vs GRE')
plt.xlabel('GRE')
plt.ylabel('CGPA')
plt.show()

data[data.CGPA >= 9].plot(kind='scatter', x='GRE Score', y='TOEFL
Score',color="GRAY",figsize=(9,6))
```

```
plt.xlabel("GRE")
plt.ylabel("TOEFL")
plt.title("CGPA>=9")

plt.show()


plt.scatter(data['CGPA'],data['SOP'])
plt.title('SOP for corresponding CGPA')
plt.xlabel('CGPA')
plt.ylabel('SOP')
plt.grid=True
plt.show()

pn = np.array([data["TOEFL Score"].min(),data["TOEFL
Score"].mean(),data["TOEFL Score"].max()])
rn = ["Worst","Average","Best"]
plt.bar(pn,rn,color="PINK")
plt.title("TOEFL Score level")
plt.xlabel("Level")
plt.ylabel("TOEFL")
plt.show()

data["GRE Score"].plot(kind = 'hist',bins = 180,color="GREEN",figsize
= (8,6))
plt.title("GRE")
plt.xlabel("GRE")
```

```
plt.ylabel("Freq")
plt.show()

gsm= np.array([data["GRE Score"].min(),data["GRE
Score"].mean(),data["GRE Score"].max()])
hsm = ["Worst","Average","Best"]
plt.bar(gsm,hsm,color="Green")
plt.title("GRE Score levels")
plt.xlabel("Level")
plt.ylabel("GRE")
plt.show()

plt.figure(figsize=(10, 10))
sns.heatmap(data.corr(), annot=True, linewidths=0.08, fmt=
'.4f',cmap="magma")
plt.show()

sns.barplot(x="University Rating", y="Chance of Admit ", data=data)

data.Research.value_counts()
sns.countplot(x="University Rating",data=data)
```

****Splitting The Data Into Train And Test****

```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.30,random
_state=10)
```

X_train.shape

X_test.shape

y_train.shape

y_train=(y_train>0.5)
y_train

y_train.shape

y_test=(y_test>0.5)
y_test

****Training and Testing the model using Logistic Regression****

from sklearn.linear_model import LogisticRegression
cls=LogisticRegression(random_state=0)
lr=cls.fit(X_train,y_train)
ypred=lr.predict(X_test)
ypred_train = lr.predict(X_train)
ypred

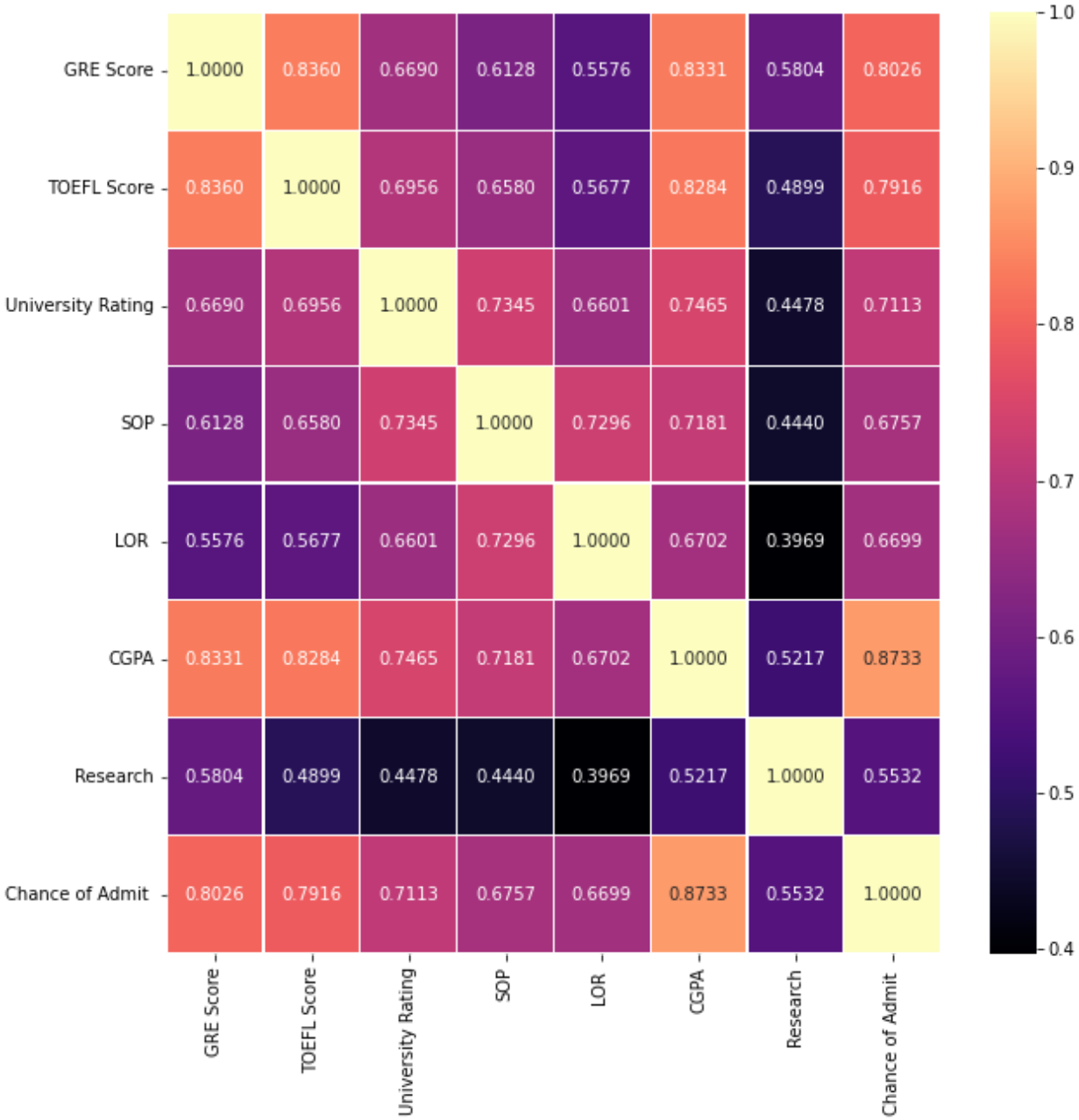
```
print("Training Accuracy : ",(metrics.accuracy_score(y_train,
ypred_train))*100)

print("Testing Accuracy : ",(metrics.accuracy_score(y_test,
ypred))*100)

from sklearn.metrics import
accuracy_score,recall_score,roc_auc_score,confusion_matrix
print("\nAccuracy score: %f" %(accuracy_score(y_test,ypred)*100))
print("Recall score : %f" %(recall_score(y_test,ypred)*100))
print("ROC score : %f\n" %(roc_auc_score(y_test,ypred)*100))
print(confusion_matrix(y_test,ypred))

import pickle
pickle.dump(lr,open('Final_deliverable.pkl','wb'))
model=pickle.load(open('Final_deliverable.pkl','rb'))
```

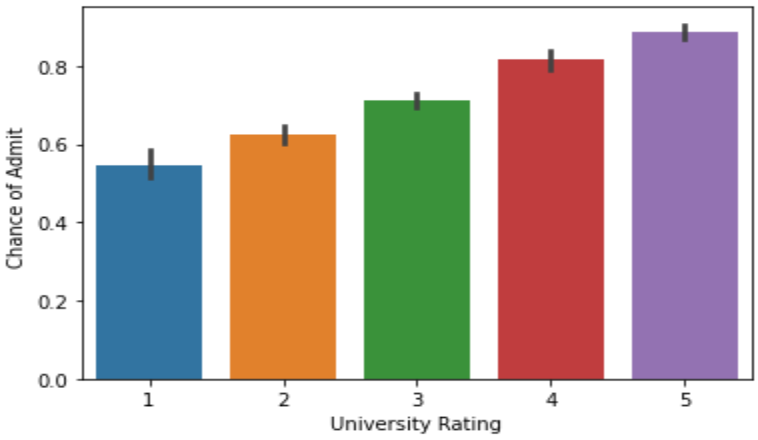
7. RESULTS AND CONCLUSION

Every year millions of students apply to universities to begin their educational life. Most of them don’t have proper resources, prior knowledge and are not cautious, which in turn creates a lot of problems as applying to the wrong university/college, which further wastes their time, money and energy. With the help of our project, we have tried to help out such students who are finding difficulty in finding the right university for them. It is very important that a candidate should apply to colleges that he/she has a good chance of getting into, instead of applying to colleges that they may never get into. This will help in reduction of cost as students will be applying to only those universities that they are highly likely to get into.Our prepared models work to a satisfactory level of accuracy and may be of great assistance to such people. This is a project with good future scope, especially for students of our age group who want to pursue their higher education in their dream college.

Results show us that the highest accuracy is achieved through the SVM Algorithm and the decision tree has the lowest accuracy.

MODEL	ACCURACY
Logistic Regression	0.93
Decision Tree	0.658
SVM Algorithm	0.744
Naive Bayes Algorithm	0.8212
KNN Algorithm	0.790


GRE Score
TOEFL Score
University Rating
SOP
LOR
CGPA
Research
Chance of Admit
1.0000 0.8360 0.6690 0.6128 0.5576 0.8331 0.5804 0.8026
0.8360 1.0000 0.6956 0.6580 0.5677 0.8284 0.4899 0.7916
0.6690 0.6956 1.0000 0.7345 0.6601 0.7465 0.4478 0.7113
0.6128 0.6580 0.7345 1.0000 0.7296 0.7181 0.4440 0.6757
0.5576 0.5677 0.6601 0.7296 1.0000 0.6702 0.3969 0.6699
0.8331 0.8284 0.7465 0.7181 0.6702 1.0000 0.5217 0.8733
0.5804 0.4899 0.4478 0.4440 0.3969 0.5217 1.0000 0.5532
0.8026 0.7916 0.7113 0.6757 0.6699 0.8733 0.5532 1.0000
1.0
0.9
0.8
0.7
0.6
0.5
0.4



University Rating vs Chance of admit

```
[ ] print("Training Accuracy : ",(metrics.accuracy_score(y_train, ypred_train))*100)

    Training Accuracy :  93.57142857142857

[ ] print("Testing Accuracy : ",(metrics.accuracy_score(y_test, ypred))*100)

    Testing Accuracy :  93.33333333333333

    from sklearn.metrics import accuracy_score,recall_score,roc_auc_score,confusion_matrix
    print("\nAccuracy score: %f" %(accuracy_score(y_test,ypred)*100))
    print("Recall score : %f" %(recall_score(y_test,ypred)*100))
    print("ROC score : %f\n" %(roc_auc_score(y_test,ypred)*100))
    print(confusion_matrix(y_test,ypred))

    Accuracy score: 93.333333
    Recall score : 99.082569
    ROC score : 67.723103

    [[  4   7]
     [  1 108]]
```

Our model giving 93.333 % of Accuracy and 99.08% of Recall Score and 67.72% of ROC Score . So this model will give the highest accuracy to our prediction.

8. REFERENCES

1. M. N. Injadat, A. Moubayed, A. B. Nassif, and A. Shami, “Systematic ensemble model selection approach for educational data mining,” Knowledge-Based Syst., vol. 200, p. 105992, Jul. 2020.

2. . S. Sujay, “Supervised Machine Learning Modelling & Analysis for Graduate Admission prediction” vol. 7, no. 4, pp. 5–7, 2020.

3. N. Chakrabarty, S. Chowdhury, and S. Rana, “A statistical Approach to Graduate Admissions’ Chance Prediction,” no. March, pp. 145– 154, 2020.

4. C. Lopez-Martin,Y. Villuendas-Rey, M. Azzeh, A. Bou Nassif, and S. Banitaan, “Transformed k-nearest neighborhood output distance minimization for predicting the defect density of software projects,” J. Syst. softw. vol. 167, p. 110592, Sep. 2020.

5. M. S. Acharya, A. Armaan, and A. S. Antony, “A comparison of regression models for Prediction of graduate admissions,” ICCIDS 2019 - 2nd Int. Conf. Comput. Intell. Data Sci. Proc., pp. 1–5, 2019.

6. M. S. Acharya, A. Armaan, and A. S. Antony, “A Comparison of Regression Models for Prediction of Graduate Admissions,” Kaggle, 2018.